Reconstructing extended Petri nets with priorities - handling priority conflicts revisited

Marie C.F. Favre, Annegret K. Wagler

Reconstructing extended Petri nets with priorities - handling priority conflicts revisited

Marie C.F. Favre [1] Annegret K. Wagler [2]

Research Report LIMOS/RR-13-14

10 février 2014

1. marie.favre@isima.fr
2. wagler@isima.fr

Abstract.

This work aims at reconstructing Petri net models for biological systems from experimental time-series data. The reconstructed models shall reproduce the experimentally observed dynamic behavior in a simulation. For that, we consider Petri nets with priority relations among the transitions and control-arcs, to obtain additional activation rules for transitions to control the dynamic behavior. An integrative reconstruction method, taking both priority relations and control-arcs into account, was proposed by Favre and Wagler in 2013. Here, we detail the aspect of choosing priorities and control-arcs such that dynamic conflicts can be resolved to finally arrive at the experimentally observed behavior.

Keywords: Petri nets, time-series data, priority relations, control-arcs

Résumé

Ce travail vise à reconstruire des modèles à l'aide de réseaux de Petri pour les systèmes biologiques à partir des séries de données expérimentales chronologiques. Les modèles reconstruits doivent reproduire le comportement dynamique observée expérimentalement lors une simulation. Pour cela, nous utilisons des réseaux de Petri associés à des relations prioritaires entre les transitions et contrôle-arcs, afin obtenir des règles d'activation supplémentaires pour les transitions et ainsi contrôler le comportement dynamique de notre modèle. Une méthode de reconstruction intégrée, prenant les deux relations prioritaires et contrôle-arcs en compte, a été proposé par Favre et Wagler en 2013. Ici, nous détaillons l'aspect de choix des priorités et contrôle-arcs ainsi que les conflits dynamiques pouvant être résolus pour finalement arriver à un modèle reproduisant les comportements observés expérimentalement.

Mots clés : Réseaux de Petri, données de séries temporelles, relations de priorités, arêtes de control

Acknowledgement.

This work was founded by the French National Research Agency, the European Commission (Feder funds) and the Région Auvergne within the LabEx IMobS^3.

1 Introduction

Systems biology aims at the integrated experimental and theoretical analysis of molecular or cellular networks to achieve a holistic understanding of biological systems and processes. To gain the required insight into the underlying biological processes, experiments are performed and experimental data are interpreted in terms of models. Depending on the biological aim, the type and quality of the available data, different types of mathematical models are used and corresponding reconstruction methods have been developed. Our work is dedicated to Petri nets which turned out to coherently model both static interactions in terms of networks and dynamic processes in terms of state changes, see *e.g.* [9,14].

In fact, a network $\mathcal{P} = (P, T, A, w)$ reflects the involved components by places $p \in P$ and their interactions by transitions $t \in T$, linked by weighted directed arcs $(p, t), (t, p) \in A$. Each place $p \in P$ can be marked with an integral number x_p of tokens defining a system state $\boldsymbol{x} \in \mathbb{Z}_+^{|P|}$, *i.e.*, we obtain $\mathcal{X} := \{\boldsymbol{x} \in \mathbb{Z}^{|P|} : x_p \geq 0\}$ as set of potential states. A transition $t \in T$ is enabled in a state $\boldsymbol{x}$ if $x_p \geq w(p, t)$ for all p with $(p, t) \in A$, we denote the set of all such transitions by $T(\boldsymbol{x})$. Switching $t \in T(\boldsymbol{x})$ yields a successor state $\text{succ}(\boldsymbol{x}) = \boldsymbol{x}'$ with $x'_p = x_p - w(p, t)$ for all $(p, t) \in A$ and $x'_p = x_p + w(t, p)$ for all $(t, p) \in A$. Dynamic processes are represented by sequences of such state changes.

Our central question is to reconstruct models of this type from experimental time-series data by means of an exact, exclusively data-driven approach. This approach takes as input a set P of places and discrete time-series data $\mathcal{X}'$ given by sequences $(\boldsymbol{x}^0, \boldsymbol{x}^1, \ldots, \boldsymbol{x}^k)$ of experimentally observed system states. The goal is to determine all Petri nets (P, T, A, w) that are able to reproduce the data, *i.e.*, that perform for each $\boldsymbol{x}^j \in \mathcal{X}'$ the experimentally observed state change to $\boldsymbol{x}^{j+1} \in \mathcal{X}'$ in a simulation. Hence, in contrast to the normally used stochastic simulation, we require that for states where at least two transitions are enabled, the decision between the alternatives is not taken randomly, but a specific transition is selected. Thus, (standard) Petri nets have to be equipped with additional activation rules to force the switching of special transitions (to reach $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$), and to prevent all others from switching. For that, different types of additional activation rules are possible.

On the one hand, in [2] the concept of control-arcs is used to represent catalytic or inhibitory dependencies. An *extended Petri net* $\mathcal{P} = (P, T, (A \cup A_R \cup A_I), w)$ is a Petri net which has, besides the (standard) arcs in A, two additional sets of so-called control-arcs: the set of read-arcs $A_R \subset P \times T$ and the set of inhibitor-arcs $A_I \subset P \times T$; we denote the set of all arcs by $\mathcal{A} = A \cup A_R \cup A_I$. Here, a transition $t \in T(\boldsymbol{x})$ coupled with a read-arc (resp. an inhibitor-arc) to a place $p \in P$ can switch only if at least $w(p, t)$ tokens (resp. less than $w(p, t)$ tokens) are present in p; we denote by $T_{\mathcal{A}}(\boldsymbol{x})$ the set of all such transitions.

On the other hand, in [12,16,18] priority relations among the transitions of a network are employed to reflect the rate of the corresponding reactions, where the fastest reaction has highest priority and, thus, is taken. In Marwan et al. [12] it is proposed to model such priorities with the help of partial orders $\mathcal{O}$ on the transitions. We call $(\mathcal{P}, \mathcal{O})$ an *(extended) Petri net with priorities*, if $\mathcal{P} = (P, T, \mathcal{A}, w)$ is an (extended) Petri net and $\mathcal{O}$ a priority relation on T. Priorities can prevent enabled transitions from switching: For each state $\boldsymbol{x}$, a transition $t \in T_{\mathcal{A}}(\boldsymbol{x})$ is allowed to switch only if there is no other enabled transition in $T_{\mathcal{A}}(\boldsymbol{x})$ with higher priority; we denote by $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ the set of all such transitions.

To enforce a deterministic behavior, $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ must contain at most one element for each $\boldsymbol{x} \in \mathcal{X}$ to enforce that $\boldsymbol{x}$ has a unique successor $\mathrm{succ}_{\mathcal{X}}(\boldsymbol{x})$, see [16] for more details. For our purpose, we consider a relaxed condition, namely that $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ contains at most one element for each experimentally observed state $\boldsymbol{x} \in \mathcal{X}'$, but $T_{\mathcal{A},\mathcal{O}}(\boldsymbol{x})$ may contain several elements for non-observed states $\boldsymbol{x} \in \mathcal{X} \setminus \mathcal{X}'$. We call such Petri nets *$\mathcal{X}'$-deterministic*.

Based on earlier results in [2,3,5,12,18], we proposed in [7] an integrative method to reconstruct all $\mathcal{X}'$-deterministic extended Petri nets with priorities fitting given experimental time-series data $\mathcal{X}'$ (see Section 2). The contribution of this paper is to detail the aspect of choosing priorities and control-arcs: we discuss the mathematical structures and underlying combinatorial problems which allow us to effectively resolve dynamic conflicts in order to finally arrive at the experimentally observed dynamic behavior (see Section 3). We close with some concluding remarks and lines of future research.

2 Reconstructing extended Petri nets with priorities

We describe the input, the main ideas, and the output of our approach from [7].

Input. A set of components P (standing for proteins, enzymes, genes, receptors or their conformational states, later represented by the set of places) is chosen which is expected to be crucial for the studied phenomenon.

To perform an experiment, one first triggeres the system in some state $\boldsymbol{x}^0$ (by external stimuli like the change of nutrient concentrations or the exposition to some pathogens), to generate an initial state $\boldsymbol{x}^1$. Then the system's response to the stimulation is observed and the resulting state changes are measured for all components at certain time points. This yields a sequence of (discrete or discretized) states $\boldsymbol{x}^j \in \mathbb{Z}^{|P|}$ reflecting the time-dependent response of the system to the stimulation in $\boldsymbol{x}^1$, which typically terminates in a terminal state $\boldsymbol{x}^k$ where no further changes are observed. The corresponding experiment is

$$\mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^k) = (\boldsymbol{x}^0; \boldsymbol{x}^1, \ldots, \boldsymbol{x}^k).$$

Several experiments starting from different initial states in a set $\mathcal{X}'_{ini} \subseteq \mathcal{X}'$, reporting the observed state changes for all components $p \in P$ at certain time points, and ending at different terminal states in a set $\mathcal{X}'_{term} \subseteq \mathcal{X}'$ describe the studied phenomenon, and yield experimental time-series data of the form

$$\mathcal{X}' = \{\mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^k) : \boldsymbol{x}^1 \in \mathcal{X}'_{ini}, \boldsymbol{x}^k \in \mathcal{X}'_{term}\}.$$

Thus, the input of the reconstruction approach is given by $(P, \mathcal{X}')$.

Example 1. As running example, we will consider experimental biological data from the *light-induced sporulation of Physarum polycephalum* as in [7,18]. In *P. polycephalum* plasmodia, the photoreceptor involved in the control of sporulation Spo is a protein which occurs in two stages P_{FR} and P_R. The developmental decision of starving *P. polycephalum* plasmodia to enter the sporulation pathway is controlled by environmental factors like visible light [15]. If the dark-adapted form P_{FR} absorbs far-red light FR, the receptor is converted into its red-absorbing form P_R, which causes sporulation [10]. If P_R is exposed to red light R, it is photoconverted back to the initial stage P_{FR}, which prevents sporulation. The experimental setting consists of

$$\begin{aligned} P = \{FR, R, P_{FR}, P_R, Spo\},\ \mathcal{X}'(\boldsymbol{x}^1, \boldsymbol{x}^3) &= (\boldsymbol{x}^0; \boldsymbol{x}^1, \boldsymbol{x}^2, \boldsymbol{x}^3),\ \mathcal{X}'_{ini} &= \{\boldsymbol{x}^1, \boldsymbol{x}^4\}, \\ \mathcal{X}'(\boldsymbol{x}^4, \boldsymbol{x}^0) &= (\boldsymbol{x}^2; \boldsymbol{x}^4, \boldsymbol{x}^0), \quad \mathcal{X}'_{term} &= \{\boldsymbol{x}^3, \boldsymbol{x}^0\} \end{aligned}$$

as input for the algorithm, we present all observed states schematically in Fig 1.

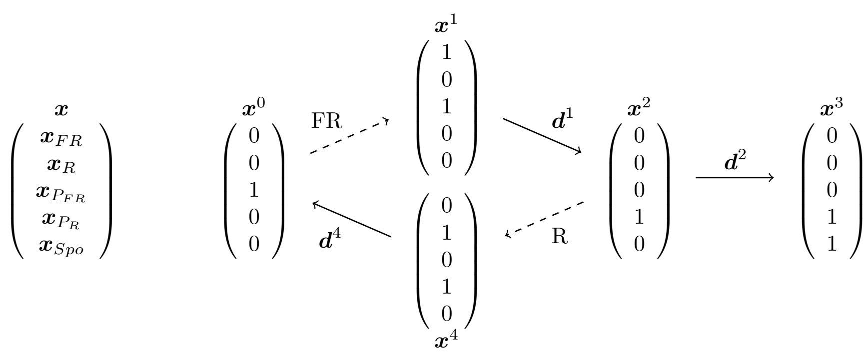

Fig. 1. A scheme illustrating the experimental time-series data described in Exp. 1 concerning the light-induced sporulation of *Physarum polycephalum*, where the entries of the state vectors are interpreted as shown on the left (dashed arrows represent stimulations $\boldsymbol{x}^0 \to \boldsymbol{x}^1$, solid arrows responses $\boldsymbol{x}^j \to \boldsymbol{x}^{j+1}$).

For a successful reconstruction, the data $\mathcal{X}'$ need to satisfy two properties: reproducibility and monotonicity.

The data $\mathcal{X}'$ are *reproducible* if for each $\boldsymbol{x}^j \in \mathcal{X}'$ there is a unique observed successor state

$$\mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) = \boldsymbol{x}^{j+1} \in \mathcal{X}'.$$

Reproducibility is obviously necessary and can be ensured by a preprocessing step [20].

Note that a state $\boldsymbol{x}^j \in \mathcal{X}'$ and its observed successor $\mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) = \boldsymbol{x}^{j+1} \in \mathcal{X}'$ are not necessarily also consecutive system states (this depends on the chosen time points to measure the states in $\mathcal{X}'$). Instead, $\boldsymbol{x}^{j+1}$ may be obtained from $\boldsymbol{x}^j$

by a switching sequence of some length, where the intermediate states are not reported in $\mathcal{X}'$. The data $\mathcal{X}'$ are *monotone* if for each pair $(\boldsymbol{x}^j, \boldsymbol{x}^{j+1}) \in \mathcal{X}'$, the values of the elements do not oscillate in the possible intermediate states between $\boldsymbol{x}^j$ and $\boldsymbol{x}^{j+1}$. It was shown in [3] that monotonicity has to be required, too (which is equivalent to demand that all essential responses are indeed reported in $\mathcal{X}'$).

Output. An extended Petri net with priorities $(\mathcal{P}, \mathcal{O})$ with $\mathcal{P} = (P, T, \mathcal{A}, w)$ fits the given data $\mathcal{X}'$ when it is able to perform every observed state change from $\boldsymbol{x}^j \in \mathcal{X}'$ to $\boldsymbol{x}^{j+1} \in \mathcal{X}'$. This can be interpreted as follows. With $\mathcal{P}$, an *incidence matrix* $M \in \mathbb{Z}^{|P|\times|T|}$ is associated, where each row corresponds to a place $p \in P$ of the network, and each column $M_{\cdot t}$ to the *update vector* $\boldsymbol{r}^t$ of a transition $t \in T$:

$$r_p^t = M_{pt} := \begin{cases} -w(p,t) & \text{if } (p,t) \in A, \\ +w(t,p) & \text{if } (t,p) \in A, \\ 0 & \text{otherwise.} \end{cases}$$

Reaching $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$ by a switching sequence using the transitions from a subset $T' \subseteq T$ is equivalent to obtain the state vector $\boldsymbol{x}^{j+1}$ from $\boldsymbol{x}^j$ by adding the corresponding columns $M_{\cdot t}$ of M for all $t \in T'$:

$$\boldsymbol{x}^j + \sum_{t \in T'} M_{\cdot t} = \boldsymbol{x}^{j+1}.$$

T has to contain enough transitions to perform all experimentally observed switching sequences. The underlying standard network $\mathcal{P} = (P, T, A, w)$ is *conformal* with $\mathcal{X}'$ if, for any two consecutive states $\boldsymbol{x}^{j+1} \in \mathcal{X}'$, the linear equation system $\boldsymbol{x}^{j+1} - \boldsymbol{x}^j = M\boldsymbol{\lambda}$ has an integral solution $\boldsymbol{\lambda} \in \mathbb{N}^{|T|}$ such that $\boldsymbol{\lambda}$ is the incidence vector of a sequence $(t^1, ..., t^m)$ of transition switches, *i.e.*, there are intermediate states

$$\boldsymbol{x}^j = \boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^{m+1} = \boldsymbol{x}^{j+1}$$

with $\boldsymbol{y}^l + M_{\cdot t^l} = \boldsymbol{y}^{l+1}$ for $1 \le l \le m$. The extended Petri net $\mathcal{P} = (P, T, \mathcal{A}, w)$ is *catalytic conformal* with $\mathcal{X}'$ if $t^l \in T_{\mathcal{A}}(\boldsymbol{y}^l)$ for each intermediate state $\boldsymbol{y}^l$, and the extended Petri net with priorities $(\mathcal{P}, \mathcal{O})$ is $\mathcal{X}'$*-deterministic* if $\{t^l\} = T_{\mathcal{A},\mathcal{O}}(\boldsymbol{y}^l)$ holds for all $\boldsymbol{y}^l$.

The desired output of the reconstruction approach consists of the set of all $\mathcal{X}'$-deterministic extended Petri nets $(\mathcal{P}, \mathcal{O})$ (all having the same set P of places as part of the input).

Example 2. We represent in Fig. 3 (page 10) the 8 alternative $\mathcal{X}'$-deterministic extended Petri nets fitting the experimental data $\mathcal{X}'$ from our running example.

We now briefly sketch the reconstruction procedure.

Representation of observed responses. As initial step, extract the observed changes of states from the experimental data. For that, define the set

$$\mathcal{D} := \left\{ \boldsymbol{d}^j = \boldsymbol{x}^{j+1} - \boldsymbol{x}^j \ : \ \boldsymbol{x}^{j+1} = \mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^j) \in \mathcal{X}' \right\}.$$

Generating the complete list of all $\mathcal{X}'$-deterministic extended Petri nets $\mathcal{P} = (P, T, \mathcal{A}, w)$ includes finding the corresponding standard networks and their incidence matrices $M \in \mathbb{Z}^{|P|\times|T|}$. Hence, the first step is to describe the set of potential columns of M. Due to monotonicity [3], it suffices to represent any $\boldsymbol{d}^j \in \mathcal{D}$ using sign-compatible vectors from the following set only:

$$\mathrm{Box}(\boldsymbol{d}^j) = \left\{ \boldsymbol{r} \in \mathbb{Z}^{|P|} : \begin{array}{r} 0 \le r_p \le d_p \text{ if } d_p^j > 0 \\ d_p \le r_p \le 0 \ \text{ if } d_p^j < 0 \\ r_p = 0 \ \text{ if } d_p^j = 0 \end{array} \right\} \setminus \{\boldsymbol{0}\}.$$

Next, we determine for any $\boldsymbol{d}^j \in \mathcal{D}$, the set $\Lambda(\boldsymbol{d}^j)$ of all integral solutions of

$$\boldsymbol{d}^j = \sum_{\boldsymbol{r}^t \in \mathrm{Box}(\boldsymbol{d}^j)} \lambda_t \boldsymbol{r}^t, \ \lambda_t \in \mathbb{Z}_+,$$

and for each $\lambda \in \Lambda(\boldsymbol{d}^j)$, the (multi-)set $\mathcal{R}(\boldsymbol{d}^j, \lambda) = \{\boldsymbol{r}^t \in \mathrm{Box}(\boldsymbol{d}^j) : \lambda_t \neq 0\}$ of update vectors used for this solution λ. By construction, $\mathrm{Box}(\boldsymbol{d}^j)$ and $\Lambda(\boldsymbol{d}^j)$ are always non-empty since $\boldsymbol{d}^j$ itself is always a solution due to reproducibility [7].

Every permutation $\pi = (\boldsymbol{r}^{t_1}, \ldots, \boldsymbol{r}^{t_m})$ of the elements of a set $\mathcal{R}(\boldsymbol{d}^j, \lambda)$ gives rise to a sequence of intermediate states $\boldsymbol{x}^j = \boldsymbol{y}^1, \boldsymbol{y}^2, ..., \boldsymbol{y}^m, \boldsymbol{y}^{m+1} = \boldsymbol{x}^{j+1}$ with

$$\sigma = \sigma_{\pi,\lambda}(\boldsymbol{x}^j, \boldsymbol{d}^j) = \big((\boldsymbol{y}^1, \boldsymbol{r}^{t_1}), (\boldsymbol{y}^2, \boldsymbol{r}^{t_2}), \ldots, (\boldsymbol{y}^m, \boldsymbol{r}^{t_m})\big)$$

which induces a priority relation $\mathcal{O}_\sigma$ since transition t_i resulting from $\boldsymbol{r}^{t_i}$ is supposed to have highest priority in $\boldsymbol{y}^i$ for $1 \le i \le m$.

Example 3. For the running example we obtain as sequences

$$\begin{tikzcd}
 & \boldsymbol{x}^0 \arrow[rrr, dashed, "\mathrm{FR}"] & & & \boldsymbol{x}^1 \arrow[dl, "\boldsymbol{r}^2"'] \arrow[dd, "\boldsymbol{d}^1"] \arrow[dr, "\boldsymbol{r}^1"] & & \\
\boldsymbol{x}^2 \arrow[ur, "\boldsymbol{r}^4"] & & \boldsymbol{x}^6 \arrow[ul, "\boldsymbol{r}^3"'] & \boldsymbol{x}^5 \arrow[dr, "\boldsymbol{r}^1"'] & & \boldsymbol{x}^0 \arrow[dl, "\boldsymbol{r}^2"] & \\
 & \boldsymbol{x}^4 \arrow[ul, "\boldsymbol{r}^3"] \arrow[uu, "\boldsymbol{d}^4"] \arrow[ur, "\boldsymbol{r}^4"'] & & & \boldsymbol{x}^2 \arrow[lll, dashed, "\mathrm{R}"] \arrow[rr, "\boldsymbol{d}^2"] & & \boldsymbol{x}^3
\end{tikzcd}$$

with $\boldsymbol{x}^5 = (1, 0, 0, 1, 0)^T$ and $\boldsymbol{x}^6 = (0, 1, 1, 0, 0)^T$.

Sequences and their conflicts. Two sequences σ and σ' are in *priority conflict* if there are update vectors $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and intermediate states $\boldsymbol{y}, \boldsymbol{y}'$ such that $t, t' \in T(\boldsymbol{y}) \cap T(\boldsymbol{y}')$ and $(\boldsymbol{y}, \boldsymbol{r}^t) \in \sigma$ but $(\boldsymbol{y}', \boldsymbol{r}^{t'}) \in \sigma'$ (since this implies $t > t'$ in $\mathcal{O}_\sigma$ but $t' > t$ in $\mathcal{O}_{\sigma'}$). We have a weak (resp. strong) priority conflict if $\boldsymbol{y} \neq \boldsymbol{y}'$ (resp. $\boldsymbol{y} = \boldsymbol{y}'$) which can (resp. cannot) be resolved by adding control-arcs.

Priority conflict graph. Construct a *priority conflict graph* $\mathcal{G} = (V_D \cup V_{term}, E_D \cup E_W \cup E_S)$ whose nodes correspond to all possible sequences $\sigma_{\pi,\lambda}(\boldsymbol{x}^j, \boldsymbol{d}^j)$ and whose edges reflect weak and strong priority conflicts:

- V_D contains the sequences $\sigma_{\pi,\boldsymbol{\lambda}}(\boldsymbol{x}^j, \boldsymbol{d}^j)$ for all $\boldsymbol{x}^j \in \mathcal{X}' \setminus \mathcal{X}'_{term}$ and the difference vector $\boldsymbol{d}^j = \mathrm{succ}_{\mathcal{X}'}(\boldsymbol{x}^i) - \boldsymbol{x}^i$, for all $\boldsymbol{\lambda} \in \Lambda(\boldsymbol{d}^j)$ and all permutations π of $\mathcal{R}(\boldsymbol{d}^j, \boldsymbol{\lambda})$.
- V_{term} contains for all $\boldsymbol{x}^k \in \mathcal{X}'_{term}$ the trivial sequence $\sigma(\boldsymbol{x}^k, \boldsymbol{0})$.
- E_D contains all edges between two sequences σ, σ' stemming from the same difference vector
- E_S (resp. E_W) reflects all SPCs (resp. WPCs) between sequences σ, σ' stemming from distinct difference vectors.

The edges in E_D induce a clique partition $\mathcal{Q}$ of V_D in as many cliques [3] as there are observed states in $\mathcal{X}' \backslash \mathcal{X}'_{term}$ resp. difference vectors in $\mathcal{D}$: $V_{\mathcal{D}} = Q_1 \cup \ldots \cup Q_{|\mathcal{D}|}$. Moreover, each node in V_{term} corresponds to a clique of size 1, so that $\mathcal{G}$ is partitioned into $|\mathcal{X}'|$ many cliques.

For illustration, we present in Fig. 2 the WPCs and SPCs between sequences of our running example.

Selection of suitable sequences. In $\mathcal{G}$, all node subsets S are generated that select exactly one sequence $\sigma_{\pi,\lambda}(\boldsymbol{x}^j, \boldsymbol{d}^j)$ per difference vector $\boldsymbol{d}^j \in \mathcal{D}$ such that no SPCs occur between the selected sequences. The set of all such solutions $S \cup V_{term}$ can be encoded by all vectors $\boldsymbol{x} \in \{0,1\}^{|V_D \cup V_{term}|}$ satisfying

$$\sum_{\sigma \in Q_j} \boldsymbol{x}_\sigma = 1 \qquad \forall Q_j \in \mathcal{Q} \tag{1a}$$

$$\boldsymbol{x}_\sigma = 1 \qquad \forall \sigma \in V_{term} \tag{1b}$$

$$\boldsymbol{x}_\sigma + \boldsymbol{x}_{\sigma'} \leq 1 \qquad \forall \sigma\sigma' \in E_S \tag{1c}$$

$$\boldsymbol{x}_\sigma \in \{0,1\} \; \forall \sigma \in V_D \cup V_{term}. \tag{1d}$$

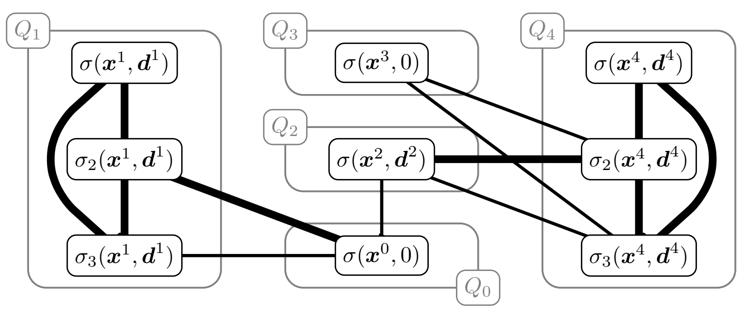

Fig. 2. The priority conflict graph resulting from the running example, where bold edges indicate SPCs, thin edges WPCs and gray boxes the clique partition $\mathcal{Q}$.

3. A clique is a subset of mutually adjacent nodes.

Example 4. From $\mathcal{G}$ shown in Fig. 2, we obtain as feasible subsets $S_i \cup V_{term}$:

$$S_1 = \{\sigma_1(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma_1(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_1(\boldsymbol{x}^4, \boldsymbol{d}^4)\},\ \ S_3 = \{\sigma_1(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma_1(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)\},$$
$$S_2 = \{\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma_1(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_1(\boldsymbol{x}^4, \boldsymbol{d}^4)\},\ \ S_4 = \{\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1), \sigma_1(\boldsymbol{x}^2, \boldsymbol{d}^2), \sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)\}.$$

Construction of standard networks, inserting control-arcs. Each subset S gives rise to a standard network $\mathcal{P}_S = (P, T_S, A_S, w)$ which is conformal with $\mathcal{X}'$ and can be made $\mathcal{X}'$-deterministic by inserting appropriate control-arcs and combining the priority relations $\mathcal{O}_\sigma \forall \sigma \in S$. Let $P(\boldsymbol{y}, \boldsymbol{y}') = \{p \in P : y_p \neq y'_p\}$:

- we obtain the columns of the incidence matrix M_S of $\mathcal{P}_S$ by taking the union of all sets $\mathcal{R}(\boldsymbol{d}^j, \lambda)$ of the sequences $\sigma = \sigma_{\pi,\lambda}(\boldsymbol{x}^j, \boldsymbol{d}^j)$ selected by $\sigma \in S$;
- for each WPC between $\sigma, \sigma' \in S$ involving update vectors $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and intermediate states $\boldsymbol{y} \neq \boldsymbol{y}'$, include either a read-arc $(p,t) \in A_R$ with weight $w(p,t) > y'_p$ for some $p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p > y'_p$ or an inhibitor-arc $(p,t) \in A_I$ with weight $w(p,t) < y_p$ for some $p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p < y'_p$ to disable transition t resulting from $\boldsymbol{r}^t$ at $\boldsymbol{y}'$,
- for each $\sigma \in S$, define $\mathcal{O}_\sigma = \{t_i > t : t \in T_{A_S \cup A_R \cup A_I}(\boldsymbol{y}^i) \setminus \{t_i\}, 1 \leq i \leq m\}$ and let $\mathcal{O}_S = \bigcup_{\sigma \in S} \mathcal{O}_\sigma$ be the studied partial order.

This implies finally that every extended network $\mathcal{P}_S = (P, T_S, A_S \cup A_R \cup A_I, w)$ together with the partial order $\mathcal{O}_S$ is $\mathcal{X}'$-deterministic, see [7] for details.

Example 5. We apply the method only to the feasible set $S_4 \cup V_{term}$ from Exp. 4. The standard network $\mathcal{P}_{S_4} = (P, T_{S_4}, A_{S_4})$ has $T_{S_4} = \{\boldsymbol{r}^1, \boldsymbol{r}^2, \boldsymbol{d}^2, \boldsymbol{r}^3, \boldsymbol{r}^4\}$. There are four WPCs between sequences of S_4:

WPC1 between	$\sigma_3(\boldsymbol{x}^1, \boldsymbol{d}^1)$	and $\sigma(\boldsymbol{x}^0, \boldsymbol{0})$	due to	$\boldsymbol{r}^2, 0$	$\in T(\boldsymbol{x}^1) \cap T(\boldsymbol{x}^0)$
WPC2 between	$\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2)$	and $\sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)$	due to	$\boldsymbol{d}^2, \boldsymbol{r}^4$	$\in T(\boldsymbol{x}^2) \cap T(\boldsymbol{x}^4)$
WPC3 between	$\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2)$	and $\sigma(\boldsymbol{x}^0, \boldsymbol{0})$	due to	$\boldsymbol{d}^2, 0$	$\in T(\boldsymbol{x}^2) \cap T(\boldsymbol{x}^0)$
WPC4 between	$\sigma(\boldsymbol{x}^3, 0)$	and $\sigma_3(\boldsymbol{x}^4, \boldsymbol{d}^4)$	due to	$0, \boldsymbol{r}^4$	$\in T(\boldsymbol{x}^3) \cap T(\boldsymbol{x}^4)$

For WPC1, we obtain $P(\boldsymbol{x}^1, \boldsymbol{x}^0) = \{FR\}$, by $\boldsymbol{x}^1_{FR} > \boldsymbol{x}^0_{FR}$, the read-arc $(FR, \boldsymbol{r}^2)$ disables $\boldsymbol{r}^2$ at $\boldsymbol{x}^0 \in \mathcal{X}'_\tau$. For WPC2, we have $P(\boldsymbol{x}^2, \boldsymbol{x}^4) = \{R\}$, by $\boldsymbol{x}^2_R < \boldsymbol{x}^4_R$, the read-arc $(R, \boldsymbol{r}^4)$ disables $\boldsymbol{r}^4$ at $\boldsymbol{x}^2$ or, alternatively, the inhibitor-arc $(R, \boldsymbol{d}^2)$ disables $\boldsymbol{d}^2$ at $\boldsymbol{x}^4$. For WPC3, we obtain $P(\boldsymbol{x}^2, \boldsymbol{x}^0) = \{P_{FR}, P_R\}$, to disable $\boldsymbol{d}^2$ at $\boldsymbol{x}^0 \in \mathcal{X}'_\tau$, by $\boldsymbol{x}^2_{P_R} > \boldsymbol{x}^0_{P_R}$, the read-arc $(P_R, \boldsymbol{d}^2)$ or, by $\boldsymbol{x}^2_{P_{FR}} < \boldsymbol{x}^0_{P_{FR}}$, the inhibitor-arc $(P_{FR}, \boldsymbol{d}^2)$ can be used. For WPC4, we have $P(\boldsymbol{x}^3, \boldsymbol{x}^4) = \{R, Spo\}$, to disable $\boldsymbol{r}^4$ at $\boldsymbol{x}^3 \in \mathcal{X}'_\tau$, by $\boldsymbol{x}^4_R > \boldsymbol{x}^3_R$, the read-arc $(R, \boldsymbol{r}^4)$ or, by $\boldsymbol{x}^4_{Spo} < \boldsymbol{x}^3_{Spo}$, the inhibitor-arc $(Spo, \boldsymbol{r}^4)$ can be used. All possible control-arcs have weight 1.

The priority relation $\mathcal{O}_4 = \{(\boldsymbol{r}^2 > \boldsymbol{r}^1), (\boldsymbol{r}^4 > \boldsymbol{r}^3)\}$ is required, the resulting 8 alternative $\mathcal{X}'$-deterministic extended Petri nets are presented in Fig. 3.

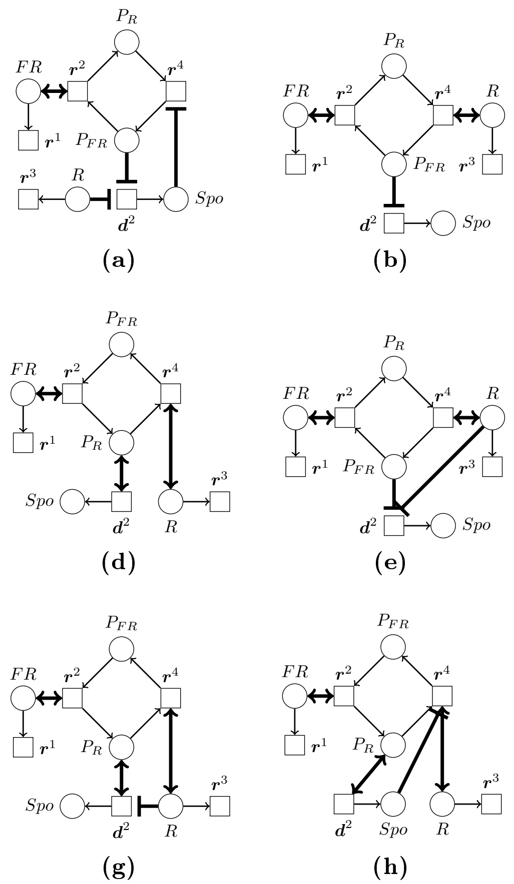

Fig. 3. The eight $\mathcal{X}'$-deterministic extended Petri nets resulting from $\mathcal{P}_{S_4}$.

3 Handling and resolving priority conflicts

In this section, we discuss the underlying mathematical structures and combinatorial problems of the two main reconstruction steps.

3.1 The priority conflict graph and selecting sequences

By construction of the priority conflict graph $\mathcal{G} = (V_D \cup V_{term}, E_D \cup E_W \cup E_S)$ and the selection of $S \subseteq V_D$ according to system (1), we note the following. Every solution $S \cup V_{term}$ of (1) corresponds to a stable set [4] in the *strong priority conflict graph* $\mathcal{G}_S = (V_D \cup V_{term}, E_D \cup E_S)$. In particular, the two constraints (1a) and (1b) enforce to select exactly one node from each of the cliques $Q_1, \dots Q_{|D|}$ and all nodes from V_{term}, resp. Since $Q_1, \dots Q_{|\mathcal{D}|}$ together with the nodes from V_{term}

4. A stable set is a subset of pairwise non-adjacent nodes.

built a clique partition $\mathcal{Q}$ of $\mathcal{G}_S$ by construction, we aim at finding a stable set $S \cup V_{term}$ of the same size as the clique cover number $\overline{\chi}(\mathcal{G}_S)$ is. In general, not every graph G has a stable set of size $\overline{\chi}(G)$, but making use of the reproducibility of the given data $\mathcal{X}'$, we can ensure that at least the selection of all sequences $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d}^j) = ((\boldsymbol{x}^j, \boldsymbol{d}^j))$ in $\mathcal{G}_S$, called canonical solution S_1, always satisfies (1). This implies:

Theorem 1. *$\mathcal{G}_S$ has at least one stable set of size $\overline{\chi}(G_S)$ and, thus, system (1) is always feasible.*

Finding all solutions of (1) means to enumerate all stables sets of size $\overline{\chi}(\mathcal{G}_S)$ in $\mathcal{G}_S$, *i.e.*, all maximum stable stets of $\mathcal{G}_S$. We next discuss which nodes of $\mathcal{G}_S$ can never enter any solution of (1) and propose a corresponding reduction of $\mathcal{G}_S$.

Lemma 1. *A node $\sigma \in V_D$ can never be selected for any solution S if there is a clique Q in $\mathcal{Q}$ so that σ is in strong conflict with all sequences $\sigma' \in Q$.*

Corollary 1. *No $\sigma \in V_D$ can be selected for any solution S if the sequence contains a terminal state as intermediate state.*

This leads to the following reduction of the priority conflict graph: We obtain the *reduced priority conflict graph* $\mathcal{G}' = (V'_\mathcal{D} \cup V_{term}, E'_D \cup E'_S \cup E'_W)$ from $\mathcal{G}$ by recursively removing from V_D all nodes which are completely joined to a clique in $\mathcal{Q}$ and the edges being adjacent to them.

Example 6. The reduced priority conflict graph $\mathcal{G}'$ of the running example can be obtained as follows: We remove $\sigma_2(\boldsymbol{x}^1, \boldsymbol{d}^1)$ since it is in SPC with (the only sequence $\sigma(\boldsymbol{x}^0, \boldsymbol{0})$) in $\mathcal{Q}_0$ due to $\boldsymbol{r}^2, \boldsymbol{0} \in T(\boldsymbol{x}^0)$. We omit $\sigma_2(\boldsymbol{x}^4, \boldsymbol{d}^4)$ since it is in SPC with (the only sequence $\sigma(\boldsymbol{x}^2, \boldsymbol{d}^2)$) in $\mathcal{Q}_2$ due to $\boldsymbol{d}^2, \boldsymbol{r}^4 \in T(\boldsymbol{x}^2)$.

Furthermore, let $\mathcal{G}'_S = (V'_\mathcal{D} \cup V_{term}, E'_D \cup E'_S)$ be the *reduced strong priority conflict graph*. We finally obtain from the above considerations:

Theorem 2. *The sets of maximum stable sets in $\mathcal{G}_S$ and $\mathcal{G}'_S$ are equal.*

Hence, we can also reduce system (1) based on $\mathcal{G}'_S$ and obtain the same solutions.

3.2 Interpretation of resolving WPCs as set cover problem

For each WPC between two sequences σ, σ', there are update vectors $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and intermediate states $\boldsymbol{y} \neq \boldsymbol{y}'$ with $t, t' \in T(\boldsymbol{y}) \cap T(\boldsymbol{y}')$ s.t. $(\boldsymbol{y}, \boldsymbol{r}^t) \in \sigma$ but $(\boldsymbol{y}', \boldsymbol{r}^{t'}) \in \sigma'$. We denote this for short by WPC(σ, σ'). This priority conflict can be solved by adding control-arcs which

- either turn $\boldsymbol{r}^t$ into a transition t which is disabled at $\boldsymbol{y}'$ (then $t > t'$ forces t to switch in $\boldsymbol{y}$ whereas only t' is enabled at $\boldsymbol{y}'$),
- or turn $\boldsymbol{r}^{t'}$ into a transition t' which is disabled at $\boldsymbol{y}$ (then $t' > t$ forces t' to switch in $\boldsymbol{y}'$ whereas only t is enabled at $\boldsymbol{y}$).

For that, consider for each WPC the set $P(\boldsymbol{y}, \boldsymbol{y}')$ of places where $\boldsymbol{y}$ and $\boldsymbol{y}'$ differ.

Remark 1. If one of $\boldsymbol{y}, \boldsymbol{y}'$ is a terminal state, say $\boldsymbol{y}'$, one of the alternatives is not possible, then t has to be disabled at $\boldsymbol{y}'$ and $t > t' = 0$ holds automatically. Note that if $\boldsymbol{y} = \boldsymbol{y}'$ then $P(\boldsymbol{y}, \boldsymbol{y}') = \emptyset$ follows which is the reason why SPCs cannot be resolved by adding control-arcs.

Let $\mathrm{CA}(\sigma, \sigma')$ be the set of all possible read-arcs that can resolve $WPC(\sigma, \sigma')$, involving $\boldsymbol{r}^t \neq \boldsymbol{r}^{t'}$ and states $\boldsymbol{y} \neq \boldsymbol{y}'$ by either disabling t at $\boldsymbol{y}'$ or t' at $\boldsymbol{y}$ then $\mathrm{CA}(\sigma, \sigma')$ contains:
- a read-arc $(p, t) \in A_R$ with weight $w(p, t) > y'_p \ \forall p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p > y'_p$,
- an inhibitor-arc $(p, t) \in A_I$ with $w(p, t) < y_p \ \forall p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y_p < y'_p$,
- a read-arc $(p, t') \in A_R$ with weight $w(p, t') > y_p \ \forall p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y'_p > y_p$,
- an inhibitor-arc $(p, t') \in A_I$ with $w(p, t') < y'_p \ \forall p \in P(\boldsymbol{y}, \boldsymbol{y}')$ with $y'_p < y_p$.

Lemma 2. *Inserting in $\mathcal{P}_S$ any non-empty subset $A' \subseteq CA(\sigma, \sigma')$ resolves the weak priority conflict $WPC(\sigma, \sigma')$.*

We next discuss which subsets of control-arcs for all WPCs are suitable to turn $\mathcal{P}_S$ into a catalytical conformal extended Petri net. On the one hand, a control-arc $(p, t) \in \mathrm{CA}(\sigma, \sigma')$ might disable t at a state in a sequence $\sigma'' \in S \setminus \sigma, \sigma'$ where t is supposed to switch. In this case, (p, t) has to be removed from $\mathrm{CA}(\sigma, \sigma')$, resulting in a reduced set $\mathrm{CA}_S(\sigma, \sigma')$. On the other hand, one control-arc may resolve several WPCs in $\mathcal{P}_S$ if the corresponding sets $\mathrm{CA}_S(\sigma, \sigma')$ intersect.

This motivates the following consideration: Introduce one variable $z_{(p,t)} \in \{0, 1\}$ for each possible control-arc $(p, t) \in \mathrm{CA}_S(\sigma, \sigma')$ for all WPCs in $\mathcal{P}_S$. Construct a 0/1-matrix A_S whose columns correspond to all those variables (resp. control-arcs) and whose rows encode the incidence vectors of the sets $\mathrm{CA}_S(\sigma, \sigma')$ for all WPCs in $\mathcal{P}_S$. Then any 0/1-solution $\boldsymbol{z}$ of $A_S \boldsymbol{z} \geq \mathbf{1}$ encodes a suitable set of control-arcs resolving all WPCs in $\mathcal{P}_S$ and, thus, a *hitting set* or *cover* of A_S.

Lemma 3. *Any cover of A_S corresponds to a set of control-arcs making $\mathcal{P}_S$ catalytical conformal with $\mathcal{X}'$.*

According to [19], we are only interested in finding minimal models fitting $\mathcal{X}'$, where minimality is interpreted in the sense that all non-minimal models contain another one also fitting the data. Based on results in [19], we can show:

Lemma 4. *Non-minimal covers of A_S yield extended Peri nets with unnecessary control-arcs and, thus, being not minimal.*

Hence, it suffices to only consider minimal covers of A_S but, for the sake of completeness, we are interested in finding all of them. The set of all minimal covers of a matrix A is called its *blocker* $b(A)$. This finally implies:

Theorem 3. *All minimal catalytical conformal extended Petri nets based on $\mathcal{P}_S$ can be obtained by computing the blocker $b(A_S)$.*

Example 7. For the feasible set $S_4 \cup V_{term}$, we obtain as matrix A_{S_4}:

	(FR,$\boldsymbol{r}^2$) $\in \mathcal{A}_R$	(P$_{FR}$, $\boldsymbol{d}^2$) $\in \mathcal{A}_I$	(P$_R$, $\boldsymbol{d}^2$) $\in \mathcal{A}_R$	(Spo,$\boldsymbol{r}^4$) $\in \mathcal{A}_I$	(R,$\boldsymbol{r}^4$) $\in \mathcal{A}_R$	(R,$\boldsymbol{d}^2$) $\in \mathcal{A}_I$
WPC1	X					
WPC2					X	X
WPC3		X	X			
WPC4				X	X	

The 15 covers of A_{S_4} are shown in the table below. The 8 $\mathcal{X}'$-deterministic extended Petri nets from Fig. 3 correspond to the 8 covers (a-h) of A_{S_4} where we chose one by one, one control-arc to solve one WPC. Note that $\mathcal{P}_e, \mathcal{P}_f, \mathcal{P}_g$ and $\mathcal{P}_h$ are not minimal since they contain unnecessary control-arcs, whereas the minimal covers from $b(A_{S_4})$ correspond to the four minimal solutions $\mathcal{P}_a, \mathcal{P}_b, \mathcal{P}_c, \mathcal{P}_d$.

	$(FR, \boldsymbol{r}^2) \in \mathcal{A}_R$	$(P_{FR}, \boldsymbol{d}^2) \in \mathcal{A}_I$	$(P_R, \boldsymbol{d}^2) \in \mathcal{A}_R$	$(Spo, \boldsymbol{r}^4) \in \mathcal{A}_I$	$(R, \boldsymbol{r}^4) \in \mathcal{A}_R$	$(R, \boldsymbol{d}^2) \in \mathcal{A}_I$
$\mathcal{P}a$	WCP1	WCP3		WCP4		WCP2
$\mathcal{P}b$	WCP1	WCP3			WCP2 and WCP4	
$\mathcal{P}c$	WCP1		WCP3	WCP4		WCP2
$\mathcal{P}d$	WCP1		WCP3		WCP2 and WCP4	
$\mathcal{P}e$	WCP1	WCP3			WCP4 and WPC2	WCP2
$\mathcal{P}f$	WCP1	WCP3		WCP4	WCP2 and WCP4	
$\mathcal{P}g$	WCP1		WCP3		WCP4 and WPC2	WCP2
$\mathcal{P}h$	WCP1		WCP3	WCP4	WCP2 and WCP4	
i	WCP1	WCP3	WCP3	WCP4	WCP2 and WCP4	
j	WCP1	WCP3	WCP3	WCP4		WCP2
k	WCP1	WCP3	WCP3		WCP2 and WCP4	
l	WCP1	WCP3	WCP3		WCP2 and WCP4	WCP2
m	WCP1	WCP3		WCP4	WCP2 and WCP4	WCP2
n	WCP1		WCP3	WCP4	WCP2 and WCP4	WCP2
o	WCP1	WCP3	WCP3	WCP4	WCP2 and WCP4	WCP2

Note that $b(A_S)$ is non-empty if and only if none of the sets $\mathrm{CA}_S(\sigma, \sigma')$ is empty. Finally, one can show that $b(A_{S_1}) \neq \emptyset$ always holds for the canonical solution S_1, so there is at least one catalytical conformal network for any given $\mathcal{X}'$. By construction, all catalytic conformal extended Petri nets based on $\mathcal{P}_S$ can be made $\mathcal{X}'$-deterministic by taking all the priorities $\mathcal{O}_\sigma$ for all $\sigma \in S$.

4 Concluding Remarks

In [7], an integrative method to reconstruct all $\mathcal{X}'$-deterministic extended Petri nets with priorities fitting given experimental time series data is proposed. We detailed here the aspect of handling priority conflicts and choosing control-arcs by discussing the underlying mathematical structures and related combinatorial problems, feasibility as well as minimality issues. For that, we interpreted

- the selection of suitable sequences from the priority conflict graph $\mathcal{G}_S$ as the problem of finding all stable sets S of size $\overline{\chi}(\mathcal{G}_S)$ to obtain all conformal standard networks $\mathcal{P}_S$ (Thm. 1);
- resolving all WPCs in a standard network $\mathcal{P}_S$ as hitting set or set cover problem involving a matrix A_S whose blocker $b(A_S)$ yields all minimal catalytic conformal extended Petri nets based on $\mathcal{P}_S$ (Thm. 3).

These interpretations in terms of two classical combinatorial problems open us the possibility to apply effective techniques known from the literature to compute the blocker of a matrix [1,6,13] or to enumerate all maximal stable sets of a graph [8,17], which include all maximum ones.

Moreover, we can ensure the existence of at least one conformal network outgoing from reproducible data (Thm. 1), since we allow the occurrence of WPCs in $\mathcal{P}_S$ which can be later resolved by inserting control-arcs. In contrast, not using control-arcs does not always result in a solution [12,5], whereas not using priorities may force the insertion of artificial control-arcs [2].

5 Concluding Remarks: Long version

In [7], an integrative method to reconstruct all $\mathcal{X}'$-deterministic extended Petri nets with priorities fitting given experimental time series data is proposed. We detailed here the aspect of handling priority conflicts and choosing control-arcs by discussing the underlying mathematical structures and related combinatorial problems, feasibility as well as minimality issues. For that, we interpreted

- the selection of suitable sequences from the priority conflict graph as the problem of finding all stable sets S of size $\overline{\chi}(\mathcal{G}_S)$ in the priority conflict graph $\mathcal{G}_S$ to obtain all conformal standard networks $\mathcal{P}_S$;
- resolving WPCs as set cover problem involving a matrix A_S encoding all possible control-arcs to resolve all WPCs in a standard network $\mathcal{P}_S$ and showed that computing the blocker $b(A_S)$ yields all minimal catalytic conformal extended Petri nets based on $\mathcal{P}_S$.

These interpretations in terms of two well-known combinatorial optimization problems open us the possibility to apply effective techniques known from the literature, *e.g.*, the classical algorithm of Berge [1] or one of its recent, more efficient variants [6,13] to compute the blocker of a matrix, or algorithms to enumerate all maximal stable sets of a graph [8,17], which include all maximum ones. Here, it could be interesting to design a specialized algorithm for enumerating all stable sets of size $\overline{\chi}(\mathcal{G})$, based on a known clique partition of $\mathcal{G}$ of the same size $\overline{\chi}(\mathcal{G})$.

Moreover, we can ensure the existence of at least one conformal network outgoing from reproducible data (Thm. 1), since we allow the occurrence of WPCs in $\mathcal{P}_S$ which can be later resolved by inserting control-arcs. In contrast, during the reconstruction of standard networks without control-arcs, *all* priority conflicts have to be excluded so that we obtain a solution outgoing from reproducible data *only if* none of the observed differences $\boldsymbol{d}^j$ is enabled at a terminal state $\boldsymbol{x}^k$ [12] (since the resulting WPC(σ, σ') between $(\boldsymbol{x}^j, \boldsymbol{d}^j) \in \sigma$ and $(\boldsymbol{x}^k, \mathbf{0}) \in \sigma'$ cannot be resolved in standard networks).

On the other hand, we always obtain an extended network being conformal with reproducible data, since a catalytical conformal extended network exists if none of the observed differences $\boldsymbol{d}^j$ starts at a terminal state $\boldsymbol{x}^j$ [2], and this property is guaranteed by the preprocessing [20] (otherwise, $\boldsymbol{x}^j$ would have two different successors $\boldsymbol{x}^j + \boldsymbol{d}^j$ and $\boldsymbol{x}^j + \mathbf{0}$).

During the reconstruction of extended networks without priorities in [2], all occuring WPCs are resolved by inserting control-arcs only: a WPC(σ, σ') between $(\boldsymbol{y}, \boldsymbol{r}^t) \in \sigma$ and $(\boldsymbol{y}', \boldsymbol{r}^{t'}) \in \sigma'$ is resolved by disabling t at $\boldsymbol{y}'$ *and* disabling t' at $\boldsymbol{y}$ (since no priorities are at hand to force the desired switch). Hence, the resulting networks may contain more control-arcs than catalytic or inhibitory dependencies indeed exist, since several switches are not controled by reaction rates (reflected by priorities), but by additional control-arcs.

In contrast, during the reconstruction of extended networks with priorities we only introduce control-arcs if the experimentally observed behavior cannot be forced by priorities alone. Moreover, only using *minimal* sets of control-arcs needed to resolve all WPCs in a network $\mathcal{P}_S$ has a further advantage: instead of firstly computing all possible solutions (in terms of all possible covers of the matrix A_S) and later removing non-minimal solutions in a postprocessing step (as described in [19]), we avoid to generate such solutions already during the reconstruction process.

Our further goal is to avoid not only generating non-minimal solutions, but also minimal solutions which are "technically correct" but would be ruled out later during a subsequent verication process to check whether the returned solutions are "biological meaningful" or contradict well-established biological pre-knowledge (*e.g.* on catalysts or inhibitors of certain reactions). This could be done by integrating further biological pre-knowledge (beyond the information given with the experimental data) into the reconstruction process.

For standard networks, we already provided an implementation of the reconstruction approach using Answer Set Programming [4]. The final goal is to provide such an implementation also for extended Petri nets with priorities and to apply the presented reconstruction approach to different biological experimental data. We indeed expect an important impact of Automatic Network Reconstruction in order to support the integrated experimental and theoretical analysis of biological systems and processes towards their holistic understanding.

In [7], an integrative method to reconstruct all $\mathcal{X}'$-deterministic extended Petri nets with priorities fitting given experimental time series data is proposed. We detailed here the aspect of handling priority conflicts and choosing control-arcs by discussing the underlying mathematical structures and related combinatorial problems: we interpreted

- the selection of suitable sequences from the priority conflict graph as the problem of finding a stable set of size $\overline{\chi}(\mathcal{G}_S)$ in $\mathcal{G}_S$ and ensured the existence of such a stable set (Thm. 1).
- resolving WPCs as set cover problem involving a matrix A_S encoding all possible control-arcs to resolve all WPCs in a standard network $\mathcal{P}_S$ and showed that computing the blocker $b(A_S)$ yields all minimal catalytic conformal extended Petri nets based on $\mathcal{P}_S$ (Thm. 3).

These interpretations in terms of two well-known combinatorial optimization problems open us the possibility to apply effective techniques known from the literature, *e.g.*, the classical algorithm of Berge [1] or one of its recent, more efficient variants [6,13], to compute the blocker of a matrix.

On the other hand, there exist algorithms to enumerate all maximal stable sets of a graph [8,11,17], which include all maximum ones. Here, it could be interesting to design a specializes algorithm for enumerating all stable sets of size $\overline{\chi}(\mathcal{G})$, based on a known clique partition of $\mathcal{G}$ of the same size $\overline{\chi}(\mathcal{G})$.

References

1. C. Berge. Hypergraphs. volume 445 of *North Holland Mathematical Library, Elsevier-North Holland, Amsterdam*, 1989.
2. M. Durzinsky, W. Marwan, and A. K. Wagler. Reconstruction of extended Petri nets from time-series data by using logical control functions. *Journal of Mathematical Biology*, 66:203–223, 2013.
3. M. Durzinsky, A. K. Wagler, and R. Weismantel. A combinatorial approach to reconstruct Petri nets from experimental data. *Lecture Notes in Comp. Sc. (special Issue CMSP 2008)*, 5307:328–346, 2008.
4. M. Durzinsky M. Ostrowski, T. Schaub and W. Marwan. Automatic network reconstruction using asp. *Th. and Practice of Logic Progr.*, 11:749–766, 2011.
5. M. Durzinsky, A. K. Wagler, and R. Weismantel. An algorithmic framework for network reconstruction. *J.Theor. Computer Science*, 412(26):2800–2815, 2011.
6. T. Eiter, K. Makino, and G. Gottlob. Computational aspects of monotone dualization: A brief survey. *Discrete Applied Mathematics*, 156(11):2035–49, 2008.
7. M. Favre and A. K. Wagler. Reconstructing $\mathcal{X}'$-deterministic extended Petri nets from experimental time-series data $\mathcal{X}'$ (extended abstract). *CEUR Workshop Proceedings (Special Issue BioPPN 2013)*, 988:45–59, 2013.
8. ÃŨ. Johansson. Nlc$_2$-decomposition in polynomial time. *Internat. J. Found. Comput. Sci.*, 11(3):373–95, 2000.
9. I. Koch and M. Heiner. Petri nets. In B. H. Junker, F. Schreiber, editor, *Biological Network Analysis*, Wiley Book Series on Bioinformatics, pages 139–179, 2007.
10. T. Lamparter and W. Marwan. Spectroscopic detection of a phytochrome-like photoreceptor in the myxomycete physarum polycephalum and the kinetic mechanism for the photocontrol of sporulation. *Photochem Photobiol*, 73:697–702, 2001.
11. K. Makino and T. Uno. New algorithms for enumerating all maximal cliques. In Torben Hagerup and Jyrki Katajainen, editors, *Algorithm Theory - SWAT 2004*, volume 3111 of *LNCS*, pages 260–272. Springer Berlin Heidelberg, 2004.
12. W. Marwan, A. K. Wagler, and R. Weismantel. A mathematical approach to solve the network reconstruction problem. *Math. Meth. of OR*, 67(1):117–132, 2008.
13. K. Murakami and T. Uno. Efficient algorithms for dualizing large-scale hypergraphs. *CoRR*, abs/1102.3813, 2011.
14. J. W. Pinney, R. D. Westhead, and G. A. McConkey. Petri net representations in systems biology. *Biochem. Soc. Tarns.*, 31:1513–1515, 2003.
15. C. Starostzik and W. Marwan. Functional mapping of the branched signal transduction pathway that controls sporulation in *Physarum polycephalum. Photochem. Photobiol.*, 62:930–933, 1995.
16. L. M. Torres and A. K. Wagler. Encoding the dynamics of deterministic systems. *Math. Methods of Operations Research*, 73:281–300, 2011.
17. S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa. A new algorithm for generating all the maximal independent sets. *SIAM J. Comput.*, 6(3):505–17, 1977.
18. A. K. Wagler. Prediction of network structure. In I. Koch, F. Schreiber, and W. Reisig, editors, *Modeling in Systems Biology*, volume 16 of *Computational Biology*, pages 309–338, 2010.

19. A. K. Wagler and J.-T. Wegener. On minimality and equivalence of Petri nets (extended abstract). *CEUR Workshop Proceedings (Special Issue CS&P 2012)*, 928:382–93, 2012.
20. A. K. Wagler and J.-T. Wegener. Preprocessing for network reconstruction: Feasibility test and handling infeasibilty (extended abstract). *CEUR Workshop Proceedings (Special Issue CS&P 2013)*, (1032):434–47, 2013.

6 Appendix

The following appendix should not be considered as part of the paper, but only as confidential supplementary information for the program committee.

6.1 The priority conflict graph and selecting sequences

For any graph G, the maximum size of a stable set in a graph, called its *stability number* $\alpha(G)$, is a lower bound for its clique cover number $\overline{\chi}(G)$ (since a stable can meet a clique in at most one node). In general, we do not have equality (*e.g.*, the chordless cycle C_5 on 5 nodes has $\alpha(C_5) = 2$ but $\overline{\chi}(C_5) = 3$).

Theorem 4. *We have $\alpha(\mathcal{G}_S) = \overline{\chi}(\mathcal{G}_S)$ for the strong priority conflict graph $\mathcal{G}_S$ and, thus, system (1) is always feasible.*

Proof of Theorem 4. Making use of the reproducibility of the given data $\mathcal{X}'$ guaranteed by the preprocessing [20], we show that the canonical solution S_1, the selection of all canonical sequences $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d}^j) = ((\boldsymbol{x}^j, \boldsymbol{d}^j)) \ \forall \ \boldsymbol{d}^j \in \mathcal{D}$, is always a solution of (1).

For that we have to verify that $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d}^j)$ is neither in strong priority conflict with any other canonical sequence $\sigma_1(\boldsymbol{x}^i, \boldsymbol{d}^i)$ for some $\boldsymbol{d}^i \in \mathcal{D} \setminus \boldsymbol{d}^j$, nor with any trivial sequence $\sigma(\boldsymbol{x}, \boldsymbol{0})$ for some $\boldsymbol{x} \in \mathcal{X}'_{term}$.

Recall that the reproducibility of $\mathcal{X}'$ guarantees that each $\boldsymbol{x}^j \in \mathcal{X}'$ has a unique successor $\boldsymbol{x}^{j+1} = \boldsymbol{x}^j + \boldsymbol{d}^j \in \mathcal{X}'$, and that this includes $\boldsymbol{d}^j \neq \boldsymbol{0} \ \forall \ \boldsymbol{d}^j \in \mathcal{D}$ (otherwise, if $\boldsymbol{d}^j = \boldsymbol{0}$, then $\boldsymbol{x}^{j+1} = \mathrm{succ}(\boldsymbol{x}^j) = \boldsymbol{x}^j + \boldsymbol{0} = \boldsymbol{x}^j$ holds and, thus, $\boldsymbol{x}^j$ has two different successors $\boldsymbol{x}^{j+1} = \boldsymbol{x}^j$ and $\mathrm{succ}(\boldsymbol{x}^{j+1}) \neq \boldsymbol{x}^j$, a contradiction).

Case 1: Consider two canonical sequences $\sigma_1(\boldsymbol{x}^i, \boldsymbol{d}^i)$ and $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d}^j)$ in priority conflict, *i.e.*, we have $\boldsymbol{d}^i \neq \boldsymbol{d}^j$ and $\boldsymbol{d}^i, \boldsymbol{d}^j \in T(\boldsymbol{x}^i) \cap T(\boldsymbol{x}^j)$.

Then $\boldsymbol{x}^i \neq \boldsymbol{x}^j$ follows from reproducibility (otherwise, $\boldsymbol{x}^i = \boldsymbol{x}^j$ would have two different successors $\boldsymbol{x}^i + \boldsymbol{d}^i \neq \boldsymbol{x}^j + \boldsymbol{d}^j$ in $\mathcal{X}'$ by $\boldsymbol{d}^i \neq \boldsymbol{d}^j$), hence the priority conflict is not strong.

Case 2: Consider a canonical sequence $\sigma_1(\boldsymbol{x}^i, \boldsymbol{d}^i)$ and a trivial sequence $\sigma(\boldsymbol{x}, \boldsymbol{0})$ for some $\boldsymbol{x} \in \mathcal{X}'_{term}$ in priority conflict, *i.e.*, we have $\boldsymbol{d}^i \in T(\boldsymbol{x})$.

We infer $\boldsymbol{x} \neq \boldsymbol{x}^i$ from the reproducibility of $\mathcal{X}'$ (otherwise $\boldsymbol{x} = \boldsymbol{x}^i$ would have two different successors $\boldsymbol{x} + \boldsymbol{d}^i \neq \boldsymbol{x} + \boldsymbol{0}$ in $\mathcal{X}'$ by $\boldsymbol{d}^i \neq \boldsymbol{0}$), hence the priority conflict is not strong.

Finally, observe that neither two canonical sequences $\sigma_1(\boldsymbol{x}^i, \boldsymbol{d})$ and $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d})$ nor two trivial sequences $\sigma(\boldsymbol{x}, \boldsymbol{0})$ and $\sigma(\boldsymbol{x}', \boldsymbol{0})$ can be in any priority conflict (since $\boldsymbol{d} = \boldsymbol{d}$ and $\boldsymbol{0} = \boldsymbol{0}$ holds, resp.). □

Theorem 1 is clearly a corrollary from the more general Theorem 4.

Proof of Lemma 1. A node $\sigma \in V_D$ being in SPC with all sequences from some $Q \in \mathcal{Q}$ is adjacent to all nodes in Q. Since exactly one node, say $\sigma' \in Q$, has to be selected from Q by (1a) or (1b), inequality (1c) forces $x_\sigma = 0$ (by $x_{\sigma'} = 1$ and $x_\sigma + x_{\sigma'} \leq 1$) due to the SPC between σ and σ'. Hence, σ cannot be selected for any solution S of (1). □

Note that Lemma 1 includes the case addressed in [2] that no sequence σ containing a terminal state $\boldsymbol{x}$ as intermediate state is appropriate: Then σ contains a reaction vector $\boldsymbol{r}^t \neq \boldsymbol{0}$ which is supposed to switch in $\boldsymbol{x}$, leading to a SPC between σ and the trivial sequence $\sigma(\boldsymbol{x}, \boldsymbol{0})$ since $t \in T(\boldsymbol{x})$ holds.

Since $\sigma(\boldsymbol{x}, \boldsymbol{0})$ is a clique $Q_{\boldsymbol{x}}$ in $\mathcal{G}_S$ of size 1 and σ is in SPC with the whole clique $Q_{\boldsymbol{x}}$, Lemma 1 implies the assertion of Corollary 1.

The reduced version of the original priority conflict graph from Fig. 2 is presented in Fig. 4.

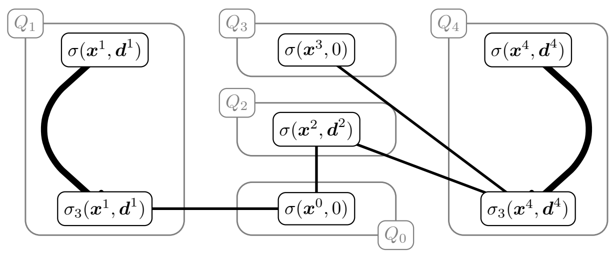

Fig. 4. The reduced priority conflict graph resulting from the running example, where bold edges indicate SPCs, thin edges WPCs and gray boxes the clique partition $\mathcal{Q}$.

Theorem 5. *We have $\alpha(\mathcal{G}_S') = \overline{\chi}(\mathcal{G}_S')$ for the reduced strong priority conflict graph and the sets of maximum stable sets in $\mathcal{G}_S$ and $\mathcal{G}_S'$ are equal.*

Proof of Theorem 5. By construction, $\mathcal{G}'$ is an induced subgraph of $\mathcal{G}$, hence $\mathcal{G}_S'$ is an induced subgraph of $\mathcal{G}_S$ as well.

Since from $\mathcal{G}$ and $\mathcal{G}_S$, only nodes are deleted during the reduction step which cannot occur in any maximum stable set of $\mathcal{G}_S$ by Lemma 1, we have $\alpha(\mathcal{G}_S') = \alpha(\mathcal{G}_S)$.

None of the cliques in $\mathcal{Q}$ turns into the empty set by applying the reduction (since at least the canonical sequence $\sigma_1(\boldsymbol{x}^j, \boldsymbol{d}^j)$ remains in Q_j and all trivial sequences $\sigma(\boldsymbol{x}, \boldsymbol{0})$ from V_{term} remain in $\mathcal{G}'$ due to Theorem 4 and Lemma 1).

This implies $\overline{\chi}(\mathcal{G}_S') = \overline{\chi}(\mathcal{G}_S)$, and the assertion $\alpha(\mathcal{G}_S') = \overline{\chi}(\mathcal{G}_S')$ follows from Theorem 4. □

Theorem 2 is a direct consequence of the more general Theorem 5.
We finally obtain the reduced system to compute all maximum stable sets in $\mathcal{G}_S'$.

$$
\begin{aligned}
\textstyle\sum_{\sigma\in Q_j} \boldsymbol{x}_\sigma &= 1 \qquad \forall Q_j \in \mathcal{Q} && \text{(2a)}\\
\boldsymbol{x}_\sigma &= 1 \qquad \forall \sigma \in V_{term} && \text{(2b)}\\
\boldsymbol{x}_\sigma + \boldsymbol{x}_{\sigma'} &\leq 1 \qquad \forall \sigma\sigma' \in E'_S && \text{(2c)}\\
\boldsymbol{x}_\sigma &\in \{0,1\}\ \forall \sigma \in V'_D \cup V_{term}. && \text{(2d)}
\end{aligned}
$$

6.2 Interpretation of resolving WPCs as set cover problem

Proof of Lemma 2. We can partition the set $\mathrm{CA}(\sigma,\sigma')$ in two different subsets:
- $\mathrm{CA}_{t,\boldsymbol{y}'}(\sigma,\sigma')$ containing all control-arcs that disable t at $\boldsymbol{y}'$,
- $\mathrm{CA}_{t',\boldsymbol{y}}(\sigma,\sigma')$ containing all control-arcs that disable t' at $\boldsymbol{y}$.

We distinguish the following cases:

Case 1: $A' \subseteq \mathrm{CA}_{t,\boldsymbol{y}'}(\sigma,\sigma')$. In this case, the control-arcs in A' disable t at $\boldsymbol{y}'$ (such that only t' remains in $T(\boldsymbol{y}')$), but still $t,t' \in T(\boldsymbol{y})$ holds. Adding the priority $t > t'$ forces t to switch in $\boldsymbol{y}$ and, thus, the $\mathrm{WPC}(\sigma,\sigma')$ is resolved by adding A' and $t > t'$.

Case 2: $A' \subseteq \mathrm{CA}_{t',\boldsymbol{y}}(\sigma,\sigma')$. This case is analogously to Case 1, by interchanging the roles of t and t' resp. $\boldsymbol{y}$ and $\boldsymbol{y}'$.

Case 3: A' intersects both $\mathrm{CA}_{t,\boldsymbol{y}'}(\sigma,\sigma')$ and $\mathrm{CA}_{t',\boldsymbol{y}}(\sigma,\sigma')$. In this case, the control-arcs in A' disable t at $\boldsymbol{y}'$ and t' at $\boldsymbol{y}$ (such that only transition t remains in $T(\boldsymbol{y})$ and transition t' in $T(\boldsymbol{y}')$). This already forces t to switch in $\boldsymbol{y}$ and t' in $\boldsymbol{y}'$, thus, the $\mathrm{WPC}(\sigma,\sigma')$ is resolved by adding A', without adding a further priority between t and t'. □

Proof of Lemma 3. Any cover C of the matrix A_S encoding row-wise all sets $\mathrm{CA}_S(\sigma,\sigma')$ for all WPCs in $\mathcal{P}_S$ selects, by construction, a non-empty subset $A' \subseteq \mathrm{CA}_S(\sigma,\sigma')$ for each WPC. According to Lemma 2, the union of these sets A' resolves all WPCs in $\mathcal{P}_S$. Moreover, it is ensured for all sequences $\sigma \in S$, that $\boldsymbol{r}^{t_l} \in T_{\mathcal{A}}(\boldsymbol{y}^l)$ holds for all reaction vectors $\boldsymbol{r}^{t_l}$ and intermediate states $\boldsymbol{y}^l$ in

$$\sigma = \sigma_{\pi,\lambda}(\boldsymbol{x}^j, \boldsymbol{d}^j) = ((\boldsymbol{y}^1, \boldsymbol{r}^{t_1}), \ldots, (\boldsymbol{y}^m, \boldsymbol{r}^{t_m})),$$

since the control-arcs do not remove t_l from $T_{\mathcal{A}}(\boldsymbol{y}^l)$ for any t_l involved in a WPC at $\boldsymbol{y}^l$ by construction of $\mathrm{CA}(\sigma,\sigma')$ and for any t_l not effected by a WPC at $\boldsymbol{y}^l$ by the reduction of the sets $\mathrm{CA}(\sigma,\sigma')$ to $\mathrm{CA}_S(\sigma,\sigma')$, if necessary.

Hence, inserting the control-arcs selected by C in $\mathcal{P}_S$ indeed yields a catalytical conformal extended Petri net. □

Note: A_S does not necessarily have a cover (namely, not if $\mathrm{CA}_S(\sigma,\sigma')$ is empty for one $\mathrm{WPC}(\sigma,\sigma')$). On the other hand, if there is a cover C for A_S, then the resulting catalytical conformal extended Petri net can be made $\mathcal{X}'$-deterministic by adding appropriate priorities (as in the proof of Lemma 2).

Proof of Lemma 4. According to [19], two $\mathcal{X}'$-deterministic extended Petri nets with priorities are M-equivalent if they have the same incidence matrix M. In particular, all $\mathcal{X}'$-deterministic extended Petri nets with priorities stemming from the same standard network $\mathcal{P}_S$ are M-equivalent (since they differ only in their sets of control-arcs or priorities).

We call a control-arc *essential* if the network is not $\mathcal{X}'$-deterministic anymore after its removal (since at least one of the WPCs in $\mathcal{P}_S$ remains unresolved), and *unnecessary* otherwise.

We ensure that a cover C of the matrix A_S encoding row-wise all sets $\mathrm{CA}_S(\sigma, \sigma')$ for all WPCs in $\mathcal{P}_S$ is minimal if and only if it only contains essential control-arcs:

Case 1: Each $\mathrm{CA}_S(\sigma, \sigma')$ intersects C in exactly one element. Then all control-arcs selected by C are clearly essential and C is a minimal cover.

Case 2: There is a set $\mathrm{CA}_S(\sigma_i, \sigma_i')$ having with C at least 2 elements in common.

Case 2.1: (At least) one control-arc in $\mathrm{CA}_S(\sigma_i, \sigma_i')$ does not show up in the intersection of C with any other $\mathrm{CA}_S(\sigma_j, \sigma_j')$, say (p_i, t_i). Then $C \setminus \{(p_i, t_i)\}$ is still a cover of A_S and, thus, C is not minimal and contains an unnecessary control-arc (p_i, t_i).

Case 2.2: All control-arcs in $\mathrm{CA}_S(\sigma_i, \sigma_i') \cap C$ show up in the intersection of C with another set $\mathrm{CA}_S(\sigma_j, \sigma_j')$. If each of them is the only control-arc in $\mathrm{CA}_S(\sigma_j, \sigma_j') \cap C$, then all of them are essential and C is minimal. Otherwise, at least one of them, say (p, t), is not the only control-arc in $\mathrm{CA}_S(\sigma_j, \sigma_j') \cap C$ (for each WPC with $(p, t) \in \mathrm{CA}_S(\sigma_j, \sigma_j')$). Then $C \setminus \{(p, t)\}$ is still a cover of A_S, C not minimal and $(p, t) \in C$ unnecessary. $\square$